**Universitat de Lleida**

**Escola Politècnica Superior**

**Màster en Enginyeria de Programari Lliure**

**Treball de Final de Màster**

# Implementació d'un sistema de votació sobre la xifra de Paillier i ElGamal

Víctor Mateu Meseguer

Lleida - Setembre de 2009

Tutors

JOSEP M. MIRET BIOSCA

FRANCESC SEBÉ FEIXAS

CAPíTOL 1

# Introducció

L'opinió és un dret que té tota persona. Cap persona hauria de prohibir a una altra opinar lliurement sobre un tema. No obstant, tothom sap que algunes opinions són més conflictives que altres. En aquest context la votació tradicional exerceix com un mitjà idoni per donar anonimat a l'opinió dels votants. Com s'aconsegueix aquest anonimat? En primer lloc tenim el sobre que conté el vot, ningú pot veure el contingut del sobre i per tant ningú sap el vot que conté. En segon lloc, abans d'obrir-se l'urna, els sobres seran barrejats, fent impossible associar sobre amb votant.

Com es pot veure, el sistema aconsegueix els objectius marcats, però encara hi ha alguns problemes. Un d'aquests problemes, potser el més evident, és la necessitat de centralitzar el lloc de votació, és a dir, tothom haurà de desplaçar-se fins l'entitat on es dugui a terme la votació. És incòmode però, en principi, tothom pot anar-hi.

Relacionat amb aquest primer entrebanc ens trobem amb el segon, la coacció. Si tothom va a votar al mateix lloc, un coaccionador maliciós pot obligar a alguns votants a votar el que ell els demani, ja que òbviament no li costa res identificar les seves víctimes. Si això succeeix, tindríem un grup de votants que no només no donarien la seva opinió, sinó que donaria més pes a l'opinió del coaccionador.

Un cop superats els entrebancs de la votació, arriba l'hora del recompte. El cost en temps de fer el recompte de vots és prou significatiu com per intentar retallar-lo dràsticament. A mès, un error en el recompte pot pasar inadvertit. Això pel que fa a temps, però també s'ha d'invertir persones en aquest recompte, concretament una quantitat que creixerà proporcionalment al tamany del cens.

La votació és, per tant, una bona manera de donar veu a tota la gent que vulgui opinar sobre un tema i es trobi dins del cens de votació. Aquí trobem un altre problema, el cens de la votació marca quines persones poden votar, però què succeeix si aquest cens és manipulat i s'afegeixen persones que no hi haurien de ser? En aquest cas la votació ja no té validesa perque estaria votant gent que no es troba al cens i estaria eliminant per tant el valor de la votació de moltes persones.

En aquesta situació, i amb la intenció d'aprofitar els avantatges que ofereixen les noves tecnologies dins del món de la votació, neix la votació electrònica. La idea és fer exactament els mateixos passos, però de manera segura, remota i automatitzada, aprofitant els beneficis de la tecnologia.

Detectat el problema de confiança dins de la votació electrònica, s'han estat realitzant estudis per trobar-hi solucions. Podem parlar de sistemes més complexos que asseguren que el vot no pot ser desxifrat per cap personatge maliciós, però el problema de cara a la societat sembla ser el mateix. Un criptosistema millor no els dóna totes les garanties que ells necessiten. Una votació no es pot simplificar només al vot d'una persona, el que interessa també és que la votació no estigui manipulada i és en aquest punt on neix la desconfiança.

Fins ara tenim que un votant envia el seu vot amb “total” seguretat a través de la xarxa, i aquest vot acaba en mans d'una entitat que en farà la permutació, el desxifratge i posterior recompte. Però com sabem que podem confiar en aquesta entitat? Segurament existeix algun certificat que l'autoritza com a "entitat confiable"però no és suficient. El que necessitem són proves de correctesa. Aquestes proves permeten demostrar que la votació no ha estat manipulada, sense necessitat de conèixer claus, ni el vot de ningú.

## 1. Objectius i Treball realitzat

El sistema [**11**] presenta una prova per demostrar la correctesa d'un procés de mescla de vots dissenyada per a treballar amb vots xifrats de Paillier. En aquest treball hem adaptat aquesta prova per treballar amb vots xifrats amb ElGamal en la seva versió additiva.

Per dur a terme el projecte s'han realitzat dues tasques paral·lelament:

- Aprenentatge:
  - Estudiar la xifra de Paillier.
  - Esudiar la xifra de ElGamal.
  - Analitzar el funcionament de la prova de correctesa presentada a [**11**].
  - Analitzar les modificacions a fer a la prova [**11**] per tal de ser utilitzada amb la xifra de ElGamal additiu
- Implementació:
  - Implementar un aplicatiu que simuli una votació electrònica.

  - Adaptar l'aplicatiu per a que pugui treballar amb qualsevol criptosistema.
  - Implementar la prova de correctesa de [**11**]
  - Implementar la variació de la prova per treballar sobre ElGamal additiu.

- Per acabar s'han analitzat els resultats obtinguts.

La memòria conté tres capítols d'explicació i un de conclusions i treball futur. En els tres primers fem un breu repàs a l'evolució de la votació electrònica, mentre que als dos següents parlem dels criptosistemes, ElGamal i Paillier concretament, explicant els conceptes bàsics que necesitarem per entendre les proves de correctesa del capítol següent.

CAPíTOL 2

# Votació electrònica

La tecnologia ha donat la possibilitat d'incrementar les capacitats de la votació tradicional en tots els sentits. Aquestes millores són tant evidents que han fet proliferar molt ràpidament els interessos en la democràcia electrònica, això provoca que es comencin a realitzar estudis per la seva legalització i n'accelera les inversions en investigació en aquest camp. Però la votació electrónica no és limita al context de les eleccions de govern, també resulta molt útil en les votacions que es duen a terme en organitzacions, tant públiques com privades, que és troben distribuïdes geograficament.

Entrant a parlar de votació electrònica, parlalem de les fases en que dividim el procés de votació, cadascuna amb uns procesos interns que pertenen simular els processos d'una votació tradicional. Aquestes fases són:

- Fase de preparació.
  (1) Anunci de l'elecció: L'autoritat de l'elecció publica el proposit de l'elecció, la llista de candidats, requisits, condicions, etc.
  (2) Registre de votants: Es busca aconseguir informació sobre els votants per a la seva posterior identificació.
- Fase de votació.
  (1) Autenticació del votant: El votant s'autentica per poder emetre el vot.

(2) Establiment de sessió: Es relaciona el votant amb el servidor de vot al que es conecta per efectuar la votació.

(3) Selecció de candidats: Es mostra la llista de candidats per a que el votant esculli.

(4) Enviament del vot: Un cop escollit el candidat, es xifra l'elecció del votant i s'envia al servidor corresponent per a la seva tramitació.

(5) Validació del vot: Es verifica que el votant només ha votat un cop i que el vot emés és correcte.

- Consolidació de resultats

(1) Transferencia de vots: Quan ha acabat la fase de votació l'autoritat de votació transfereix els vots a l'autoritat d'escrutini.

(2) Permutació de vots: S'utilitza una funció de permutació de vots per trencar qualsevol lligam entre el votant i el seu vot. S'acostuma a combinar amb un rexifratge.

(3) Desxifratge de vots: Es recupera el vot original per al seu posterior recompte.

(4) Escrutini.

(5) Auditoria de resultats: Comprovació que els resultats són correctes i que no hi ha hagut cap problema tant en la recolecció com en el recompte.

(6) Publicació de resultats.

Hi diferents tipus de votació electrònica. Amb el temps se n'han anat creant de nous a mesura que les noves tecnologies i els coneixements de seguretat ho han permés.

## 1. Tipus

En primer lloc parlarem, per antiguitat, del sistema de votació basat en paper però amb recompte automàtic. Aquest va ser la primera evolució del sistema de vot normal en què la gent porta el seu vot dins d'un sobre, que serà dipositat en una urna després de comprovar que el votant es troba dins del cens. El que pretenia aquest sistema de votació era facilitar el compte de vots fent-lo automatitzat. La idea era utilitzar sistemes de recompte de vots mitjançant l'escaneig òptic i/o tabulació electrònica, de manera que els votants podien emplenar la seva butlleta manualment i després la màquina en faria el recompte ràpidament.

Més endavant la tecnologia va evolucionar fins al punt de permetre als votants generar les seves propies butlletes amb una màquina. Aquest sistema de votació tot i ser força efectiu, tampoc representava una evolució remarcable en el món de la votació ja que l'únic problema que resol és el del recompte però segueix forçant a les persones a desplaçar-se.

Un altre tipus de votació és la votació electrònica de registre directe. En aquest cas el votant es troba amb un dispositiu, semblant a un petit ordenador, que li permetrà autenticar-se i realitzar el seu vot directament, sense butlleta ni urna. La idea és fer més accesible el vot a totes les persones. Un cop enregistrats tots el vots, aquests queden guardats dins de la memòria del dispositiu, des d'on es poden enviar a qualsevol sistema de centralització de vots. Normalment el dispositiu compta els vots a mesura que els votants els van inserint, de manera que quan els

ha d'enviar, els envia ja comptats i deixa una còpia en memòria per una possible revisió. Els dispositius han anat evolucionant a mesura que s'han trobat defectes en el sistema, i avui per avui ja disposen de sistemes de seguretat tant en la transferencia de vots, com en la recolecció, per evitar manipulacions no desitjades en els recomptes. En aquestes sistemes, la necessitat de desplaçar-se per part del votant no s'ha solucionat.

Una evolució de l'anterior, és la votació electrònica de registre directe utilitzant una xarxa pública. Ara els dispositius envien la informació a través d'Internet. Serà important que les dades es transmetin de forma segura i per tant, s'afegiran mecanismes de xifratge i signatura digital als dispositius per assegurar que la votació ha estat feta correctament. Aquest tipus de votació ja s'ha utilitzat durant força temps en les votacions internes de les empreses i també en votacions dins de departaments d'àmbit públic.

Per últim, tenim la votació electrònica per xarxa pública, en la que els votants disposen de maquinari per poder fer la votació utilitzant Internet. Sense cap dubte és la més complicada en quant a seguretat, però també la més favorable al votant ja que no li cal desplaçar-se. És en aquest camp en el què aprofundirem més. El repte és aconseguir una votació des de qualsevol lloc amb un nivell de seguretat encara més elevat que en el cas de la votació tradicional. S'ha de tenir en compte que s'haurà d'utilitzar un software específic per dur a terme aquestes votacions i per tant caldrà assegurar-se que no només els nostres sistemes informàtics són segurs, sinó que el votant ho està fent sense cap possibilitat de ser "espiat".

## 2. Requisits de seguretat

La principal sospita que recau sobre un procés electrònic és la seva seguretat en front d'un procés que es realitza manualment. Per tant qualsevol possibilitat d'intervenció no desitjada en el procés ha de ser controlada, tant si prové de fora com de dintre del propi sistema.

Els següents punts constitueixen una descripció del requisits de seguretat que ha de complir una votació electrònica:

**Legitimitat del votant:** Només poden participar votants autoritzats i només es tindrà en compte un vot per votant. Normalment s'utilitza un cens per identificar els votants autoritzats. En el cas de la votació electrònica s'utilitzen tècniques d'identificació remota.

**Robustesa:** Un votant només pot realitzar un vot i ningú fora del cens pot efectuar una votació.

**Confidencialitat:** La relació entre el votant i el seu vot no pot ser coneguda ni dedüible.

**Precisió:** El resultat de la votació ha de venir donat pel recompte de tots els vots recollits de manera legítima. S'ha d'evitar qualsevol alteració dels vots mitjançant sistemes de prevenció i detecció d'alteracions.

**No informació:** No es poden conèixer resultats parcials durant la votació ja que el coneixement de l'estat de la votació podria influir en els votants que encara no han votat.

**Verificació individual:** El votant ha de poder comprovar que el seu vot ha estat enviat, rebut i processat correctament.

**Verificació universal:** És important que hi hagi una comprovació pública per a que qualsevol participant i/o observador pugui verificar la integritat del resultat.

**Evitar coercions:** Un votant no hauria de poder demostrar a un tercer el vot que ha escollit.

CAPíTOL 3

# Criptografia

Des dels temps de l'Antic Egipte fins avui dia, ha passat molt de temps, però hi ha coses que no han canviat, com és el desig de l'ésser humà d'amagar els seus secrets.

Personatges com Cleopatra o César van aprendre la importancia d'amagar els seus missatges de les mirades indiscretes. La Scilata que els espartans utilitzaven prop del 400 a.C, o el propi codi César van ser els principis. El 1466 León Battista Alberti va idea el sistema polialfabètic basat en la rotació d'uns "corrons". Un segle més tard Giovani Battista Belaso va inventar la clau criptogràfica basada en una paraula o text que es trancrivia lletra a lletra sobre el missatge original.

El naixement de la informàtica i dels criptosistemes informàtics va suposar un canvi radical del concepte de criptografia, i també del criptoanàlisi. Els criptosistemes i els algoritmes van augmentar considerablement la seva complexitat. Des del DES fins als criptosistemes asimètrics de corbes el·líptiques hi ha hagut molts canvis.

Criptografia, l'art d'amagar, la paraula té el seu origen en el greg: kryptos (amagat) i graphein (escriure). L'art d'amagar un missatge mitjançant signes convencionals és molt antic, gairebé tant com l'escriptura. Claude E.Shannon va publicar en dos anys, dos documents que van suposar la fundació de la Teoria de la Informació [**14**], [**15**].

La criptografia es complementa amb una altra branca d'estudi, el criptoanàlisi, que estudia el camí invers de la criptografia. Dedica els seus esforços a desenmascarar els secrets que la criptografia intenta amagar.

En el context criptogràfic, considerem com a text en clar qualsevol informació que resulta llegible i compensible. Un text en clar seria qualsevol informació abans de ser encriptada o després de ser desencriptada. Es considera que qualsevol informació és vulnerable si es troba en aquest estat. En aquest mateix context, considerem com a criptograma, qualsevol informació que es trobi convenienment xifrada, i no resulti llegible ni comprensible més que per al destinatari legítim de la mateixa.

Al mecanisme per transformar un text en clar en un criptograma l'anomenen xifratge. De la mateixa manera anomenem xifratge al procés de recuperar la informació a partir d'un criptograma.

En un criptosistema la informació segueix sempre un mateix flux.

(1) L'emissor xifra el text en clar, i envia el criptograma resultant.

(2) El criptograma es transmés per un canal insegur fins arribar al receptor.

(3) El criptograma arriba al receptor, que el desxifra i obté el text en clar.

Hi ha diferents tipus de criptosistemes:

**Algoritmes simètrics:** Són els criptosistemes més senzills. Es tracta d'algoritmes que treballen amb una única clau amb doble funció. Dins d'aquests criptosistemes podem distingir entre dos tipus d'algoritmes: els de

xifrat de bloc, i els que xifren bloc a bloc. Tot i no parlar de la implementació dels algoritmes, si que citarem els principals algoritmes criptogràfics simètrics:

- DES [**16**] (Data Encryption Standard ): Algoritme de 64 bits de clau, dels quals 56 componen la clau del xifrat, mentre que els 8 restants son de paritat i s'utilitzen per corregir errors. Actualment DES ja no és estàndar criptogràfic, va ser trencat al gener de 1999, amb un sistema de còmput que analitzava 250.000.000.00 claus per segon.

- Triple DES [**1**]: Degut a la capacitat de còmput actual i la relativa facilitat que suposa trencar el DES, es desenvolupa un sistema de triple aplicació de l'algoritme DES, amb 3 claus diferents per aplicar succesivament, de fet s'utilitza una clau externa dividida, ja que DES matemàticament no és un grup, i la seva aplicació repetida provocaria un increment efectiu del tamany. Amb aquest sistema s'obté un xifrat de 192 bits, 168 efectius i 24 de paritat.

- AES (Rijndael) [**8**] (Advanced Encryption Standard): Es tracta d'un algoritme simètric de xifrat de blocs de llongitud variable. Es serveix de claus de llongitud variable: 128, 192 o 256 bits.

- IDEA [**3**] (International Data Encryption Algorithm): Va ser creat al 1990 per X. Lai i L.Massey. Es tracta d'un algoritme simètric de xifrat en blocs de 64 bits. El seu funcionament es basa en operacions senzilles com multiplicacions d'enters, sumes i XOR. Treball amb claus de 128 bits.

**Funcions hash:** Els algoritmes de resum, coneguts també com a funcions o algoritmes hash, constitueixen un tipus especial de funcions criptogràfiques. Molts manuals de criptografia els situen com un subgrup dels criptosistemes simètrics, però els considerem un grup independent per les seves característiques especials. En els algoritmes de hash no existeix el concepte de clau criptogràfica, ni tampoc el concepte de desxifrat, mentre que apareix un nou conecepte anomenat resum o hash. S'utilitza per al càlcul de codis d'autenticació i en signatura digitial.

Les característiques més importants d'un hash són:

- Unidireccional: Conegut un hash és computacionalment impossible la reconstrucció del missatge original.
- Compressió: A partir d'un missatge de qualsevol longitud s'obté un hash de tamany fix.
- Difussió: El resum és una funció complexa de tots els bits del missatge.
- Col·lissió simple: Donat un missatge qualsevol, és computacionalment impossible trobar un altre missatge, el resum del qual sigui el mateix.
- Col·lissió forta: És computacionalment dificil trobar dos missatges amb un resum idèntic.

Alguns dels principals algoritmes criptográfics de resum són:

- MD5 [**13**] (Message Digest 5): Va ser ideat pel matemàtic Ron Rivest el 1992, i suposa l'evolució dels algoritmes MD2 i MD4. Es tracta d'una funció criptogràfica de tipus hash que accepta com a entrada

un missatge de qualsevol tamany i retorna com a sortida una cadena de 128 bits.

- SHA-1 [**9**] (Secure Hash Algorithm - 1): Va ser ideat pel NIST el 1994 com una aplicació de l'algoritme SHA. Es tracta d'una funció criptogràfica de tipus hash que accepta una entrada de $2^{64}$ bits com a màxim, i retorna una cadena de 160 bits. És lleugerament més lent que MD5 però també és computacionalment més complex de trencar i per tant més segur.

**Algoritmes asimètrics:** Són criptosistemes més moderns i complexos que els simètrics, i per tant més segurs. Es basa en l'existència d'un parell de claus complementaries de manera que un criptograma xifrat per una de les claus només pot ser desxifrat per l'altra clau.

Serà amb aquests últims amb els que treballarem en l'ámbit del projecte, concretament els criptosistemes que utilitzem són: Paillier i ElGamal, que expliquem a continuació.

## 1. La xifra de Paillier

El sistema criptogràfic de Paillier és un algoritme probabilístic utilitzat en la criptografia de clau pública. La seva seguretat es basa en que el problema computacional de trobar l'enèsim residu és computacionalment difícil.

L'esquema és un criptosistema homomòrfic additiu, per tant el, criptosistema ens ofereix unes característiques molt específiques com són:

- La múltiplicació de dos texts xifrats es desxifrarà com la suma dels texts en clar:

$$D(E(m,r) * E(m',r') \pmod{N^2}) = m + m' \pmod{N}.$$

- Un text xifrat elevat a la potencia d'un text en clar es desxifrarà com la multiplicació dels texts en clar:

$$D(E(m,r)^{m'} \pmod{N^2}) = m \cdot m' \pmod{N}.$$

Per més informació sobre el criptosistema, consultar [**10**].

### 1.1. Inicialització.

Per inicialitzar el sistema necessitem generar les claus, tant la pública com la privada. Per fer-ho seguim els següents passos:

(1) Escollim 2 nombres primers grans, aleatòriament i independents l'un de l'altre, tals que compleixin que $mcd(p \cdot q, (p-1), (q-1)) = 1$.

(2) Calculem $N = p \cdot q$.

(3) Calculem $\lambda = mcm(p-1, q-1)$.

(4) Escollim un enter aleatori $g \in \mathbb{Z}^*_{N^2}$

(5) Ens assegurem que $N$ divideix l'ordre de $g$ comprovant l'existència de la següent multiplicació modular inversa:

$$\mu = (L(g^{\lambda} \pmod{N^2}))^{-1} \pmod{N}.$$

On la funció L ve definida per:

$$L(u) = \frac{u-1}{N}.$$

Un cop fet això, podem donar la clau pública $(N, g)$ mentre que guardarem la clau privada $(\lambda, \mu)$ en secret.

### 1.2. Xifratge.

En aquest mateix context tenim $m$ que és el missatge a xifrar, on $m \in \mathbb{Z}_N$. Per xifrar el missatge seguirem els següents passos:

(1) Seleccionem una $r$ aleatòria tal que $r \in \mathbb{Z}_N^*$.

(2) Calculem el missatge xifrat $c$ mitjançant la funció $c(m, r) = g^m \cdot r^N \pmod{N^2}$.

Com es pot veure, la quantitat de missatges xifrats diferents que poden correspondre a un mateix missatge és $r$. Per tant tot i que tinguem un mateix missatge xifrat diversos cops, com és el cas d'una votació, un atacant no podria relacionar els missatges xifrats entre si.

#### 1.2.1. *Desxifratge.*

Donat un missatge xifrat $c$, tal que $c \in \mathbb{Z}_{N^2}^*$. Obtenim el missatge desxifrat $m$ a partir de la funció:

- $m = L(c^\lambda \pmod{N^2}) \cdot \mu \pmod{N}$.

Fixem-nos en que per desxifrar no es té en compte el valor de $r$, que era la component aleatòria del nostre text xifrat. Aquesta propietat ens ajudarà a emmascarar els nostres vots.

### 1.3. Emmascarament.

Considerem l'emmascarament, com la propietat d'un criptosistema per aconseguir, a partir d'un missatge xifrat $c$, un altre missatge xifrat $c'$ tal que al desxifrar $c$ i $c'$ el missatge obtingut sigui el mateix. Per aconseguir-ho aprofitarem el fet que treballem amb un criptosistema additiu homomòrfic on es cumpleixen les propietats següents:

- Donat un missatge xifrat $c(m, r) = g^m \cdot r^N \pmod{N^2}$, si $m = 0$ llavors $c(0, r) = r^N \pmod{N^2}$.
- Donat un missatge xifrat $c(m, r) = g^m \cdot r^N \pmod{N^2}$ i un altre missatge xifrat $c'(m', r') = g^{m'} \cdot r'^N \pmod{N^2}$ podem calcular el missatge xifrat $c'' = c \cdot c'$. Al resultat de desxifrar $c''$ li direm $m''$ tal que $m'' = L((c \cdot c')^\lambda \pmod{N^2}) \cdot \mu \pmod{N}$ i es compleix la propietat que $m'' = m + m'$.

La idea és obtenir un $c'' \neq c$ tal que $m'' = m$. Per fer-ho necessitem un c' on $m' = 0$. Per tant $c'' = c \cdot c' = g^m \cdot r^N \cdot r'^N \pmod{N^2}$.

Així doncs per emmascarar un missatge xifrat $c$ el que farem serà:

(1) Seleccionem una $r$ aleatòria tal que $r \in \mathbb{Z}_N^*$.

(2) Generem el missatge emmascarat $c'$ com $c' = c \cdot r^N \pmod{N^2}$.

## 2. La xifra de ElGamal

El sistema de xifratge de ElGamal és un algoritme de xifratge de clau pública. La seguretat de l'algoritme es basa en la suposició que calcular un logaritme discret té una complexitat computacional molt alta.

El procediment de xifratge/desxifratge està basat en càlculs sobre qualsevol grup cíclic $G$. Això fa que la seguretat del procediment depengui directament de la dificultat de calcular un logaritme discret a $G$. Per una informació més completa consultar [**5**] o [**2**].

### 2.1. Inicialització.

Per inicialitzar el sistema necessitem generar les claus, tant la pública com la privada. Per fer-ho seguim els següents passos:

(1) Generem una descripció eficient d'un grup cíclic multiplicatiu $G$ d'ordre $q$ amb un generador $g$ .

(2) Escollim una $x$ aleatoria de l'interval $1, ..., q-1$.

(3) Calculem $y = g^x$.

(4) Publiquem $y$ juntament amb la descripció de $G$. En el nostre cas escollim $p = 2 \cdot q + 1$ on $p$ i $q$ són primeres. Treballarem en un subgrup d'ordre $q$ de $\mathbb{Z}_p^*$ generat per un generador $g$.

(5) La clau privada serà la $x$ que hem escollit al pas (2).

### 2.2. Xifratge.

Per xifrar un missatge $m$ seguirem els passos següents:

(1) Escollim una $r$ aleatòria entre $1, ..., q-1$ i calculem $c_2 = g^r \pmod{p}$.

(2) Convertim el missatge m en un element $m'$ de $G$.

(3) Calculem $c_1 = m' \cdot y^r \pmod{p}$.

(4) El text xifrat correspon a la tupla $(c_1, c_2)$.

En aquest cas el text xifrat directament és una tupla.

### 2.3. Desxifratge.

Per desxifrar el text xifrat $(c_1, c_2)$ seguirem els passos següents:

(1) Calculem $y^r$ fen el següent càlcul $y^r = c_2^x$

(2) Calculem $m' = \frac{c_1}{y^r} \pmod p$

Tot seguit demostrem que el funcionament és correcte:

$$m' = \frac{c_1}{y^r} = \frac{m' \cdot y^r}{y^r} = m'$$

### 2.4. Emmascarament.

Per reemmascarar un criptograma de ElGamal el que haurem de fer és aprofitar les propietats següents:

- Donat un missatge xifrat $c = (m \cdot y^r, g^r)$, si $m = 1$, llavors $c = (y^r, g^r)$.
- Donat un missatge xifrat $c = (m \cdot y^r, g^r)$ i un altre missatge xifrat $c' = (m' \cdot y^{r'}, g^{r'})$, obtenim el missatge xifrat $c''$ com el producte component a component de les tuples anteriors. Al resultat de desxifrar $c''$ li direm $m''$ tal que $m'' = \frac{m \cdot y^r \cdot m' \cdot y^{r'}}{y^r \cdot y^{r'}}$ i es compleix la propietat que $m'' = m \cdot m'$.

Aprofitem aquestes dues propietats per generar un reemmascarat $c''$ a partir de $c$, que compleixi que $m'' = m$. Per fer-ho utilitzarem un $c'$ amb missatge $m' = 1$, de manera, que $c = (m \cdot y^r \pmod p, g^r \pmod p)$ i $c' = (y^{r'}, g^{r'})$.

$$c'' = c \cdot c' = (m \cdot y^r \cdot y^{r'}, g^r \cdot g^{r'}) = (m \cdot y^{r+r'}, g^{r+r'})$$

.

El resultat de desxifrar $c''$ serà:

$$m'' = \frac{m \cdot y^r \cdot y^{r'}}{y^r \cdot y^{r'}} = m$$

## 3. ElGamal additiu

Com podem veure, la xifra de ElGamal permet combinar dos texts xifrats de manera que el criptograma obtingut correspon al producte dels texts en clar originals, és a dir, és un homomorfisme multiplicatiu. Per a realitzar una implementació de la prova de correctesa [**11**] utilitzant la xifra de ElGamal necessitem dotar-lo de la propietat de ser un homomorfisme additiu. Per fer-ho xifrarem i desxifrarem amb alguns matissos.

Ens interessa que es compleixin les propietats següents:

- La múltiplicació de dos texts xifrats es desxifrarà com la suma dels texts en clar:

$$D(E(m,r) * E(m',r') \pmod p) = m + m' \pmod p.$$

- Un text xifrat elevat a la potencia d'un text en clar es desxifrarà com la multiplicació dels texts en clar:

$$D(E(m,r)^{m'} \pmod p) = m \cdot m' \pmod p.$$

### 3.1. Inicialització.

Per inicialitzar el sistema seguirem els mateixos passos que amb ElGamal explicat anteriorment.

(1) Generem una descripció eficient d'un grup cíclic multiplicatiu $G$ d'ordre $q$ amb un generador $g$ .

(2) Escollim una $x$ aleatoria entre $1, ..., q-1$.

(3) Calculem $y = g^x$.

(4) Publiquem $y$ juntament amb la descripció de $G$, $p$, $q$ i $g$ com a clau pública.

(5) La clau privada serà la $x$ que hem escollit al pas (2).

### 3.2. Xifratge.

Per xifrar un missatge $m$ seguirem els passos següents:

(1) Escollim una $r$ aleatoria entre $0, ..., q-1$ i calculem $c_2 = g^r \pmod{p}$.

(2) Convertim el missatge m en un enter $m'$ de $G$.

(3) Calculem $c_1 = g^{m'} \cdot y^r \pmod{p}$.

(4) El text xifrat correspon a la tupla $(c_1, c_2)$.

L'única modificació és que en aquest cas xifrem $g^{m'}$ en comptes de $m'$

### 3.3. Desxifratge.

Per desxifrar el text xifrat $(c_1, c_2)$ seguirem els passos següents:

(1) Calculem $y^r$ a partir de l'expressió$y^r = c_2^x \pmod{p}$.

(2) Calculem $g^{m'} = \frac{c_1}{y^r} \pmod{p}$.

(3) Busquem en una taula pregenerada el valor de $m'$.

Com estem en un context de votació electrònica, la taula que troba el valor de $m'$ no té perquè ser molt gran. En altres contexts on $m'$ pot prendre un ventall de valors molt gran, la construcció d'aquesta taula seria un problema.

### 3.4. Emmascarament.

Per reemmascarar un criptograma de ElGamal additiu, aprofitarem les següents propietats:

- Donat un missatge xifrat $c = (g^m \cdot y^r, g^r)$, si $m = 0$, llavors $c = (y^r, g^r)$.
- Donat un missatge xifrat $c = (g^m \cdot y^r, g^r)$ i un altre missatge xifrat $c' = (g^{m'} \cdot y^{r'}, g^{r'})$ obtenim el missatge xifrat c" del producte $c'' = c \cdot c'$. Al desxifrar $c''$ obtenim $g^{m+m'}$. Per tant el resultat del desxifratge és $m'' = m + m'$.

Per generar el reemmascarat $c''$ a partir de $c$ utilitzarem un $c'$ amb missatge $m' = 0$, de manera que $c = (g^m \cdot y^r, g^r)$ i $c' = (y^{r'}, g^{r'})$.

$$c'' = c \cdot c' = (g^m \cdot y^r \cdot y^{r'}, g^r \cdot g^{r'})$$

.

El resultat de desxifrar $c''$ serà:

$$g^{m''} = \frac{g^m \cdot y^r \cdot y^{r'}}{y^r \cdot y^{r'}} = g^m$$

Per tant el text en clar obtingut serà $m$.

CAPíTOL 4

# Proves de correctesa d'una mescla de vots

En l'apartat 1.1.2 hem parlat dels requisits de seguretat que havia de tenir una votació electrònica. La prova de correctesa es trobaria ubicada en la definició de verificació universal. El que es pretén és que una persona o entitat tingui la possibilitat de verificar que el procés de votació s'ha dut a terme correctament. Nosaltres ens centrem en un sistema de votació on els vots són mesclats abans de desxifrar-se- La prova de correctesa s'usa per demostrar que els vots no s'han modificat durant aquest procés.

Hi ha diferents tipus de proves de correctesa. Una distinció important és si la prova permet la comprovació a partir d'informació de la barreja, o si, per contra, la prova es realitza sense que l'auditor obtingui cap tipus d'informació sobre els vots. Aquesta distinció és molt important en tant que afecta al grau de complicitat que hi ha entre l'entitat responsable de la votació i l'entitat responsable de l'auditoria de resultats. Si la prova es realitza sense donar cap informació associada a la votació, llavors l'entitat auditora pot ser qualsevol persona ja que en cap cas rebrà informació corresponent a la votació. Com a molt rebrà el nombre de votants, que és, de fet, una dada pública dins d'una votació.

## 1. Prova de correctesa de K. Peng, C. Boyd i E. Dawson

El treball [**11**] proposa una prova de correctesa dissenyada per provar la correctesa d'una mescla de vots xifrats amb la xifra de Paillier.

L'entitat de votació és l'entitat que obté tots els vots, els barreja, i els emmascara per fer-ne l'escrutini posterior. Definim entitat auditora com la persona o conjunt de persones que volen comprovar la prova de correctesa sobre la votació de l'entitat de votació. Definim $c_i$ com els vots xifrats que ha rebut l'entitat de votació, mentre que $c'_i$ són els vots permutats i reemmascarats.

La prova que presentem tot seguit serveix per demostrar que el pas de $c_i$ a $c'_i$ s'ha fet correctament.El procediment per realitzar la prova és el següent:

(1) L'entitat de votació escull aleatòriament $r_i \in \mathbb{Z}_N^*$ per $i = 1, 2, ..., n$ i publica $c'_i = c_{\pi(i)} \cdot r_i^N \pmod{N^2}$ per $i = 1, 2, ..., n$. $n$ representa el nombre de vots de la mescla i $\pi$ fa referència a la permutació.

(2) L'entitat auditora publica $s_i \in 0, 1, ..., 2^L - 1$ per $i = 1, 2, .., n$. On $L$ és un paràmetre de seguretat.

(3) L'entitat de votació escull $r'_i \in \mathbb{Z}_N^*$ per $i = 1, 2, ..., n$ i publica $c''_i = c_i'^{t_i} \cdot r_i'^N \pmod{N^2}$ per $i = 1, 2, ..., n$ on $t_i = s_{\pi(i)}$. i publica la prova ZK següent:

$$ZP(t_i, r_i \mid c''_i = c_i'^{t_i} \cdot r_i'^N \pmod{N^2})$$

i

$$ZP(R_1 \mid R_1^N = C_1 \pmod{N^2})$$

on $R_1 = \prod_{i=1}^n r_i^{t_i} \cdot r'_i \pmod{N^2}$ i $C_1 = \frac{\prod_{i=1}^n c''_i}{\prod_{i=1}^n c_i'^{s_i}} \pmod{N^2}$ .

(4) L'entitat auditora publica $s'_i \in 0, 1, ..., 2^L - 1$ per $i = 1, 2, .., n$ Després l'entitat de votació genera $t'_i = s'_{\pi(i)}$ per $i = 1, 2, ..., n$ i publica la prova ZK següent:

$$ZP(R_2, R_3, t'_i \mid C_2 \cdot R_2^N = \prod_{i=1}^n c_i''^{t'_i} \pmod{N^2}, C_3 \cdot R_3^N = \prod_{i=1}^n c_i''^{t'_i} \pmod{N^2})$$

on $R_2 = \prod_{i=1}^{n} r_i^{t'_i} \pmod{N^2}$ , $R_3 = \prod_{i=1}^{n} r_i^{t_i \cdot t'_i \pmod{\phi)(N)} \cdot r' t'_{ii}} \pmod{N^2}$ , $C_2 = \prod_{i=1}^{n} c_i^{s'_i} \pmod{N^2}$ i $C_3 = \prod_{i=1}^{n} c_i^{s_i \cdot s'_i \pmod{\phi)(N)}} \pmod{N^2}$ .

(5) L'entitat de votació genera aleatòriament $W_1 \in \mathbb{Z}_N^*$, $W_2 \in \mathbb{Z}_N^*$, $W_3 \in \mathbb{Z}_N^*$, $v_i \in \mathbb{Z}_N$ , $v'_i \in \mathbb{Z}_N$ i $x_i \in \mathbb{Z}_N^*$ per $i = 1, 2, ..., n$.

(6) L'entitat de votació calcula $a_i = c_i'^{v_i} \cdot x_i^N \pmod{N^2}$ per $i = 1, 2, ..., n$ , $F = W_1^N \pmod{N^2}$, $A = \frac{\prod_{i=1}^{n} c_i'^{v'_i}}{W_2^N} \pmod{N^2}$ i $B = \frac{\prod_{i=1}^{n} c_i''^{v'_i}}{W_3^N} \pmod{N^2}$.

(7) L'entitat de votació calcula $c = H(F, A, B, a_1, a_2, ..., a_n)$ on $H$ és una funció hash amb una sortida de 128 bits.

(8) L'entitat de votació calcula $z_1 = W_1 \cdot R_1^c \pmod{N^2}$, $z_2 = \frac{W_2}{R_2^c} \pmod{N^2}$, $z_3 = \frac{W_3}{R_3^c} \pmod{N^2}$, $\alpha_i = x_i \cdot r_i^c \pmod{N^2}$, $\gamma_i = v_i + c \cdot t_i \pmod{N}$ , $\gamma'_i = c \cdot t'_i - v'_i \pmod{N}$.

(9) L'entitat de votació pública $z_1, z_2, z_3, \alpha_1, \alpha_2, ..., \alpha_n, \gamma_1, \gamma_2, ..., \gamma_n, \gamma'_1, \gamma'_2, .., \gamma'_n$ de manera que l'entitat auditora pot comprovar que:

$$c = H(\frac{z_1^N}{C_1^c}, \frac{C_2^c}{z_2^N \cdot \prod_{i=1}^{n} c_i'^{\gamma'_i}}, \frac{C_3^c}{z_3^N \cdot \prod_{i=1}^{n} c_i''^{\gamma'_i}}, \frac{c_1'^{\gamma_1} \cdot \alpha_1^N}{c_1''^c}, ..., \frac{c_n'^{\gamma_n} \cdot \alpha_n^N}{c_n''^c})$$

## 2. Adaptació de la prova per utilitzar-se sobre ElGamal additiu

Hem transformat la prova de [**11**] per a treballar sobre vots xifrats amb la xifra de ElGamal additiu. El resultat ha estat el següent:

(1) L'entitat de votació escull aleatòriament $r_i \in \mathbb{Z}_p^*$ per $i = 1, 2, ..., n$ i publica $c'_i = (c_{i,1} \cdot y^{r_i}, c_{i,2} \cdot g^{r_i})$ per $i = 1, 2, ..., n$.

(2) L'entitat auditora publica $s_i \in 0, 1, ..., 2^L - 1$ per $i = 1, 2, .., n$.

(3) L'entitat de votació escull $r'_i \in \mathbb{Z}_p^*$ per $i = 1, 2, ..., n$ i publica $c''_i = (c'^{t_i}_{i,1} \cdot y^{r'}, c'^{t_i}_{i,2} \cdot g^{r'}) \pmod p$ per $i = 1, 2, ..., n$ on $t_i = s_{\pi(i)}$. i publica la prova ZK

$$ZP(t_i, r_i \mid c''_i = (c'^{t_i}_{i,1} \cdot y^{r'}, c'^{t_i}_{i,2} \cdot g^{r'}) \pmod p)$$

i

$$ZP(R_1 \mid y^{R_1} = C_1 \pmod p)$$

on $R_1 = \sum_{i=1}^{n}(r_i \cdot t_i + r'_i) \pmod q$ i $C_1 = \frac{\prod_{i=1}^{n} c''_i}{\prod_{i=1}^{n} c'^{s_i}_i} \pmod p$.

(4) L'entitat auditora publica $s'_i \in 0, 1, ..., 2^L - 1$ per $i = 1, 2, .., n$ Desrpés l'entitat de votació genera $t'_i = s'_{\pi(i)}$ per $i = 1, 2, ..., n$ i publica la prova ZK:

$$ZP(R_2, R_3, t'_i \mid C_2 \cdot y^{R_2} = \prod_{i=1}^{n} c'^{t'_i}_i \pmod p, C_3 \cdot y^{R_3} = \prod_{i=1}^{n} c'''^{t'_i}_i \pmod p)$$

On $R_2 = \sum_{i=1}^{n} r_i \cdot t'_i \pmod q$ , $R_3 = \sum_{i=1}^{n} r_i \cdot t_i \cdot t'_i + r'_i \cdot t'_i \pmod p$ , $C_2 = \prod_{i=1}^{n} c_i^{s'_i} \pmod p$ i $C_3 = \prod_{i=1}^{n} c_i^{s_i \cdot s'_i \pmod q} \pmod p$.

(5) L'entitat de votació genera aleatòriament $W_1 \in \mathbb{Z}_p^*$, $W_2 \in \mathbb{Z}_p^*$, $W_3 \in \mathbb{Z}_p^*$, $v_i \in \mathbb{Z}_p$ , $v'_i \in \mathbb{Z}_p$ i $x_i \in \mathbb{Z}_p^*$ per $i = 1, 2, ..., n$.

(6) L'entitat de votació calcula $a_i = c'^{v_i}_{i,1} \cdot y^{x_i} \pmod p$ per $i = 1, 2, ..., n$ , $F = y^{W_1} \pmod p$, $A = \frac{\prod_{i=1}^{n} c'^{v'_i}_{i,1}}{yW_2} \pmod p$ i $B = \frac{\prod_{i=1}^{n} c''^{v'_i}_{i,1}}{y^{W_3}} \pmod p$.

(7) L'entitat de votació calcula $c = H(F, A, B, a_1, a_2, ..., a_n)$ on $H$ és una funció hash amb una sortida de 128 bits.

(8) L'entitat de votació calcula $z_1 = W_1 + R \cdot c_1 \pmod p$ , $z_2 = W_2 - R \cdot c_2 \pmod p$ , $z_3 = W_3 - R \cdot c_3 \pmod p$ , $\alpha_i = x_i + r_i \cdot c \pmod q$ , $\gamma_i = v_i + c \cdot t_i \pmod q$ i $\gamma'_i = c \cdot t'_i - v'_i \pmod q$.

(9) L'entitat de votació pública $z_1, z_2, z_3, \alpha_1, \alpha_2, ..., \alpha_n, \gamma_1, \gamma_2, ..., \gamma_n, \gamma'_1, \gamma'_2, .., \gamma'_n$ de manera que l'entitat auditora pot comprovar que:

$$c = H(\frac{y^{z_1}}{C_1^c}, \frac{C_2^c}{y^{z_2} \cdot \prod_{i=1}^n c_{i,1}^{\prime\gamma'_i}}, \frac{C_3^c}{y^{z_3} \cdot \prod_{i=1}^n c_{i,1}^{\prime\prime\gamma'_i}}, \frac{c_{1,1}^{\prime\gamma_1} \cdot y^{\alpha_1}}{c_{1,1}^{\prime\prime c}}, ..., \frac{c_{n,1}^{\prime\gamma_n} \cdot y^{\alpha_n}}{c_{n,1}^{\prime\prime c}})$$

### 2.1. Demostració.

Per demostrar que la prova és correcta, necessitem demostrar que:

$$c = H(F, A, B, a_1, a_2, ..., a_n) = H(\frac{y^{z_1}}{C_1^c}, \frac{C_2^c}{y^{z_2} \cdot \prod_{i=1}^n c_{i,1}^{\prime\gamma'_i}}, \frac{C_3^c}{y^{z_3} \cdot \prod_{i=1}^n c_{i,1}^{\prime\prime\gamma'_i}}, \frac{c_{i,1}^{\prime\gamma_i} \cdot y^{\alpha_i}}{c_{i,1}^{\prime\prime c}}).$$

(1) Demostrem que $F = \frac{y^{z_1}}{C_1^c}$, sabent que $F = y^{W_1} \pmod p$ i que $y^{R_1} = C_1 \pmod p$ :

$$F = \frac{y^{z_1}}{C_1^c} = \frac{y^{W_1 + R \cdot c_1}}{C_1^c} = \frac{y^{W_1} \cdot (y^{R_1})^c}{C_1^c} = \frac{y^{W_1} \cdot C_1^c}{C_1^c} = y^{W_1} \pmod p.$$

(2) Demostrem que $A = \frac{C_2^c}{y^{z_2} \cdot \prod_{i=1}^n c_{i,1}^{\prime\gamma'_i}}$ sabent que $A = \frac{\prod_{i=1}^n c_{i,1}^{\prime v'_i}}{yW_2} \pmod p$ i que $C_2 \cdot y^{R_2} = \prod_{i=1}^n c_i^{\prime t'_i} \pmod p$:

$$A = \frac{C_2^c}{y^{z_2} \cdot \prod_{i=1}^n c_{i,1}^{\prime\gamma'_i}} = \frac{C_2^c}{y^{W_2 - R \cdot c} \cdot \prod_{i=1}^n c_{i,1}^{\prime\gamma'_i}} = \frac{y^{R_2 \cdot c} \cdot C_2^c}{y^{W_2} \cdot \prod_{i=1}^n c_{i,1}^{\prime\gamma'_i}} = \frac{\prod_{i=1}^n c_{i,1}^{\prime v'_i}}{yW_2} \pmod p.$$

(3) Demostrem que $B = \frac{C_3^c}{y^{z_3} \cdot \prod_{i=1}^n c_{i,1}^{\prime\prime\gamma_i'}}$ sabent que $B = \frac{\prod_{i=1}^n c_{i,1}^{\prime\prime v_i'}}{y^{W_3}} \pmod p$ i que $C_3 \cdot y^{R_3} = \prod_{i=1}^n c_i^{\prime\prime t_i'} \pmod p$ :

$$B = \frac{C_3^c}{y^{z_3} \cdot \prod_{i=1}^n c_{i,1}^{\prime\prime\gamma_i'}} = \frac{C_3^c}{y^{W_3 - R\cdot c} \cdot \prod_{i=1}^n c_{i,1}^{\prime\prime\gamma_i'}} = \frac{y^{R_3\cdot c} \cdot C_3^c}{y^{W_3} \cdot \prod_{i=1}^n c_{i,1}^{\prime\prime\gamma_i'}} = \frac{\prod_{i=1}^n c_{i,1}^{\prime\prime v_i'}}{y^{W_3}} \pmod p.$$

(4) Demostrem que $a_i = \frac{c_{i,1}^{\prime\gamma_i} \cdot y^{\alpha_i}}{c_{i,1}^{\prime\prime c}}$ per $i = 1, 2, ..., n$ sabent que $a_i = c_{i,1}^{\prime v_i} \cdot y^{x_i} \pmod p$:

$$a_i = \frac{c_{i,1}^{\prime\gamma_i} \cdot y^{\alpha_i}}{c_{i,1}^{\prime\prime c}} = \frac{c_{i,1}^{\prime v_i + c\cdot t_i} \cdot y^{x_i + r_i\cdot c}}{c_{i,1}^{\prime\prime c}} = \frac{c_{i,1}^{\prime v_i + c\cdot t_i} \cdot y^{x_i + r_i\cdot c}}{c_{i,1}^{\prime c\cdot t_i} \cdot y^{r_i\cdot c}} = c_{i,1}^{\prime v_i} \cdot y^{x_i} \pmod p.$$

Per $i = 1, 2, ..., n$.

CAPíTOL 5

# Detalls d'implementació

Hem implementat les proves utilitzant Java com a llenguatge de programació. Bàsicament l'hem escollit per coneixements previs i perquè disposa d'eines de xifratge implementades que ens podien resultar molt útils.

El problema que teniem amb les implementacions de java és que les classes que ens ofereix el llenguatge xifren i desxifren utilitzan els algoritmes que els demanem, però no ens dóna accés ni al valor de les claus ni als valors aleatoris que es generen durant el xifratge. Aquesta situació ens obliga a generar tot el codi, però sí que aprofitarem la classe BigInteger que ens permetrà treballar amb claus suficientment grans com per a que les proves siguin significatives.

Hem aprofitat aquesta decisió per a generar un disseny de l'aplicació que ens permet optimitzar la generació de proves. A grans trets podem dividir l'implemenació en 3 parts:

**Entitat de votació:** S'encarrega de simular el funcionament d'un col·legi electoral. Principalment s'encarrega de permutar els vots rebuts, reenmascarar-los i realitzar les proves de correctesa quan li siguin requerides.

**Criptosistemes:** Ens hem encarregat de generar un disseny que ens permeti cambiar entre qualsevol criptosistema sense necessitat de fer un programa nou per cadascun. Per aconseguir-ho creem la interfície criptosistema que implementaran tots els criptosistemes que generem. D'aquesta

manera si algú vol generar proves amb un altre criptosistema només ha d'implementar-lo com a subclasse de criptosistema i fer la prova. Els criptosistemes són els encarregats de xifrar i desxifrar i permeten a cada autoritat conèixer només les dades que li son necessàries.

**Proves de correctesa:** En aquest cas no hem volgut generar una interficie ja que les proves de correctesa no tenen perquè estructurar-se de la mateixa manera tot i ser del mateix criptosistema. Per tant el criptosistema coneixerà la prova.

CAPíTOL 6

# Comparativa de resultats

La prova de [**11**] ha estat implementada sobre la xifra de Paillier i sobre la xifra de ElGamal additiu. Les dades que ens interesen són les que ens permetran comparar el rendiment de les dues versions. Per veure el creixement en temps dels algoritmes hem realitzat proves amb diferents quantitats de vots, i també amb diferents tamanys de clau. Així hem pout veure com es comporta cada algoritme i extreure'n conclusions.

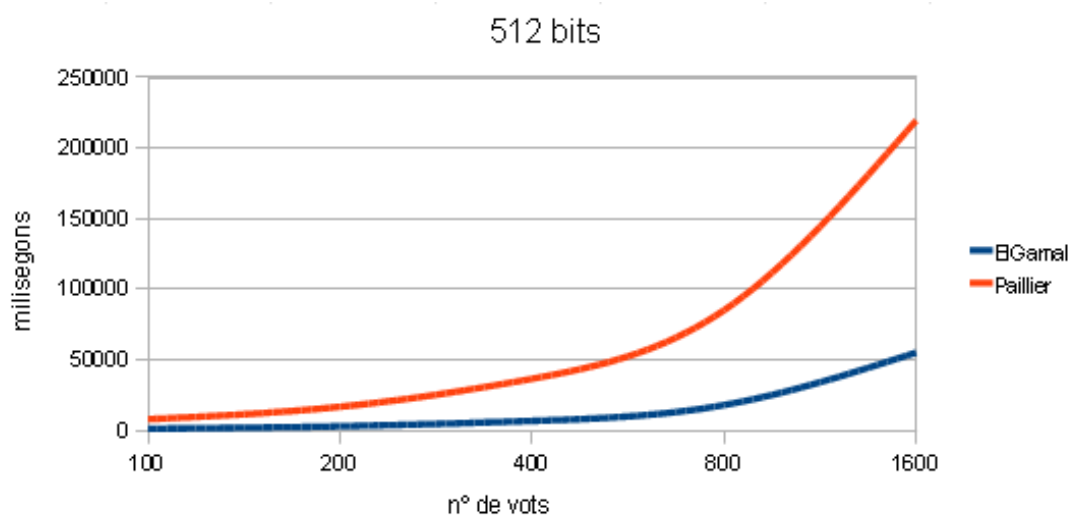


FIGURA 1. Comparativa amb clau de 512 bits

Amb una clau de 512 bits la millora del ElGamal additiu respecte Paillier és molt evident. A mesura que s'incrementa el nombre de vots s'accentua més la diferència, i en les nostres proves arriba a ser fins a 4 vegades més ràpid amb només 1600 vots.

Hem de comprovar si a mesura que incrementem el tamany de la clau, els resultats se segueixen comportant igual o si les distàncies es veuen modificades. Anem doncs a veure els resultats de les mateixes proves però ara amb una clau de 768 bits.

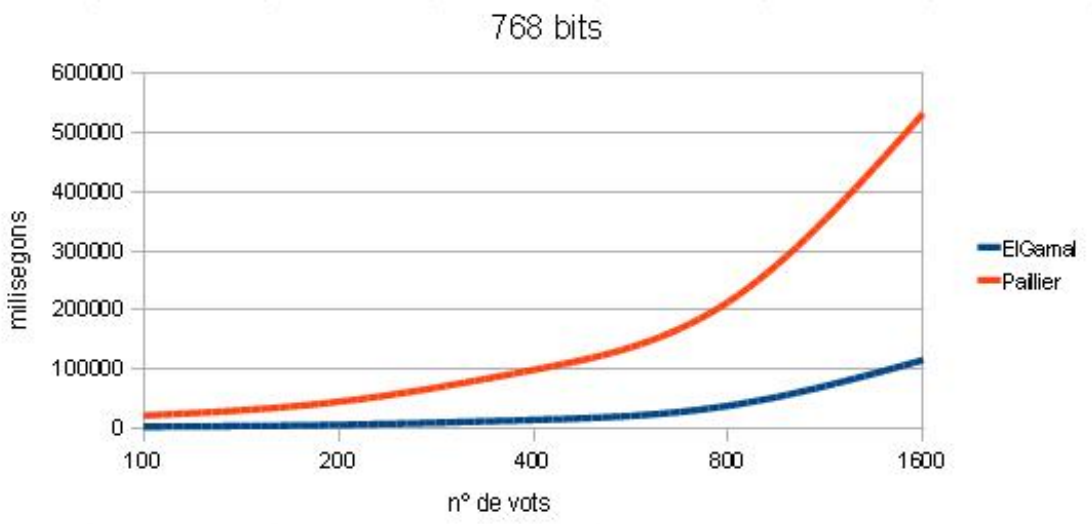


FIGURA 2. Comparativa amb clau de 768 bits

No sembla que les diferències s'hagin escurçat, més aviat s'han incrementat lleugerament. El comportament segueix sent el mateix. A mesura que s'incrementa el nombre de vots, la millora en rendiment de ElGamal és va fent més evident. Estem parlant de proves de correctesa 5 vegades més ràpides amb ElGamal que amb Paillier.

Per últim, veiem que els resultats és mantenen amb una clau de 1024 bits.

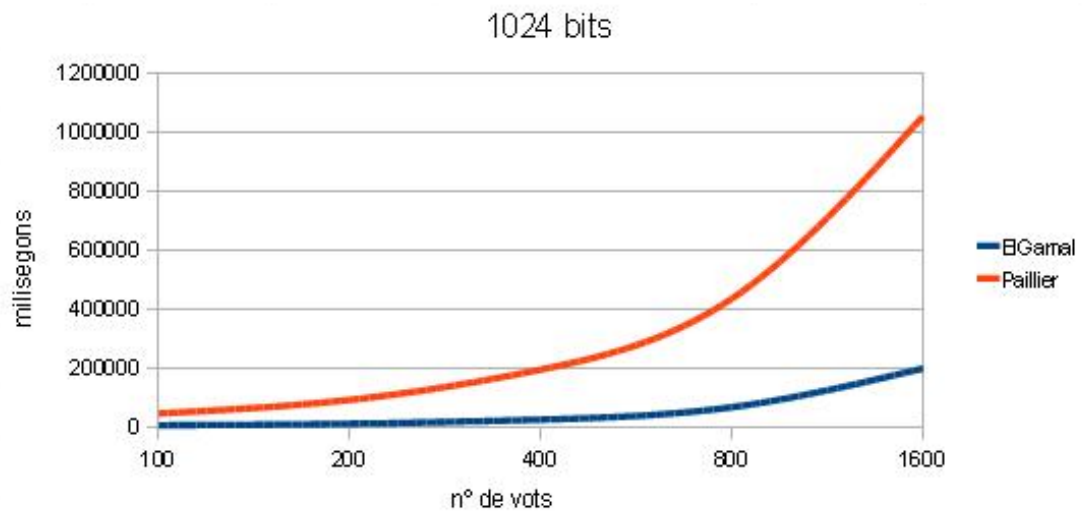


FIGURA 3. Comparativa amb clau de 1024 bits

CAPíTOL 7

# Conclusions

Després de veure els resultats obtinguts a l'hora de realitzar les proves, la conclusió més evident és que la prova que presenta [**11**] utilitzant la xifra de Paillier és menys eficient que la seva adaptació amb la xifra de ElGamal additiu.

El motiu de les seves diferències crec que és el tamany dels nombres que es generen durant la prova. Notem que quan utilitzem la xifra de Paillier l'algoritme treballa sobre $\pmod{N^2}$ mentre que amb la xifra de ElGamal additiu estem treballant sobre $\pmod{p}$, on $N$ i $p$ s tenen el mateix nombre de bits. Llavors quan utilitzem com a modul $N^2$ estem treballant amb nombres que dupliquen el nombre de bits i creiem que aquesta diferència que s'arrossega durant tota la prova acaba marcant distancies prou importants.

També hem de tenir en compte que la xifra de Paillier ens dóna uns avantatges que no estem aprofitant en les proves. La més interesant és el rang de vots que pot manejar. La xifra de Paillier pot desxifrar de forma ràpida qualsevol missatge $m \in \mathbb{Z}_N^*$. En canvi, ElGamal additiu ha de treballar amb valors de $m$ que es puguin tabular. Això en limita molt el ventall de possibilitats. En les nostres proves, els missatges a xifrar podien ser nombres de 1 a 10, entorn que en el cas d'una votació pot ser suficient, però en altres situacions no. Hem de ser conscients que com més gran és la varietat de vots, més dificultats tindrem per treballar amb ElGamal additiu.

## 1. Treball futur

- En la línea del treball realitzat es podria intentar trobar una adaptació de la xifra de Paillier que fos més competitiva en quant a temps. Buscariem sobretot que l'adaptació ens permetés treballar $\pmod{N}$ per poder analitzar la millora en els temps de càlcul de la prova.
- Posar a prova els limits de ElGamal additiu, utilitzant-lo per xifrar missatges de text no tabulables, i comprovar que el seu funcionament deixa de ser eficient.
- Aprofitar el codi de l'aplicació que hem generat per afegir nous xifratges i desenvolupar noves proves de correctesa i comparar-les.
- Dissenyar una prova de correctesa diferent i provar-la enfront de les dues proves que ja tenim implementades.
- Paral·lelitzar la prova.

# Bibliografia


[1] American National Standards Institute, *Triple Data Encryption Algorithm Modes of Operation* ANSI X9.52-1998.

[2] T. ElGamal, *A public key cryptosystem and a signature scheme based on discrete logarithms.* IEEE trans. Inform. Theory, 31, no. 4 , 469-472, 1985.

[3] X. Lai, *On the design and security of block ciphers*, ETH Series in Information Processing, J.L. Massey (editor), vol. 1, Hartung-Gorre Verlag Konstanz, Technische Hochschule (Zurich), 1992.

[4] D. Master, *Criptosistemas informáticos*, 2004.

[5] A. J. Menezes, P.C. v. Oorschot, S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press New York, 1997

[6] V. Morales, *Seguridad en los procesos de voto electrònico remoto* UPC - Departamento de Telemàtica.

[7] R. Moreno, J. Pujolàs, P. Sanz, M. Serio. *Mix verificables con pares ElGamal y curvas elípticas.* Actas VI Jornadas de Matemtica Discreta y Algorítmica, 469-476, 2008.

[8] National Institute of Standards and Technology. FIPS Pub 197: *Advanced Encryption Standard (AES).* November 2001

[9] National Institute of Science and Technology, Federal Information Processing Standard (FIPS) *Secure Hash Standard*, 180-1, April 1993.

[10] P. Paillier *Public-key Criptosystems Based on Composite Degree Residuosity Classes* Procs of EUROCRYPT'99 , LNCS vol.1592, pp. 223-238, 1999.

[11] K. Peng, C. Boyd, E. Dawson, *Simple and Efficient Shuffling with Provable Correctness and ZK Privacy* Procs of CRYPTO'05 , LNCS vol.3621, pp. 188-204, 2005.

[12] A. Riera *Design of Implementable Solutions for Large Scale Electronic Voting Schemes* PhD. UAB, 1999.

[13] R. Rivest, *The MD5 Message-Digest Algorithm*, RFC1321, MIT LCS and RSA Data Security, Inc., April 1992.

[14] C. E. Shannon, *A mathematical theory of communication*, Bell System Tech. J. 27 (1948), 379-423 y 623-656

[15] C. E. Shannon, *Communication theory of secrecy systems*, Bell System Tech. J. 28 (1949), 656-715

[16] US National Bureau of Standards, *Data Encryption Standard*, Federal Information Processing Standard (FIPS) Publication 46, January 1977